Association for Information Systems
AIS Electronic Library (AISeL)

2017 Proceedings | Portugal (CAPSI)

2017

Historical Data Management in Big Databases

José Pedro Simão
Universidade do Minho, zepedro.simao@gmail.com

Orlando Belo
Universidade do Minho, obelo@di.uminho.pt

Gestão de Dados Históricos em Bases de Dados de Grandes Dimensões

Historical Data Management in Big Databases

José Pedro Simão, Departamento de Informática, Escola de Engenharia, Universidade do Minho
Campus de Gualtar, Portugal, zepedro.simao@gmail.com
Orlando Belo, Centro ALGORITMI, Escola de Engenharia, Universidade do Minho
Campus de Gualtar, Portugal, obelo@di.uminho.pt

Resumo

É muito provável que grande parte dos sistemas de informação tenha problemas de gestão de informação. Isto obriga à criação de novos tipos de técnicas de gestão de dados mais eficientes e específicas para cada caso, com capacidade para governar e assegurar o cumprimento das medidas de gestão definidas para os sistemas, e garantir o desempenho e a qualidade desejada. Neste trabalho, abordamos o problema da gestão de dados, e, através de uma solução baseada em técnicas de *machine learning*, tentamos perceber, aprender e classificar os dados contidos numa qualquer base de dados, de acordo com a sua relevância para os utilizadores. Conseguir identificar aquilo que é realmente importante para o utilizador e separar esta informação da restante, é uma excelente forma de diminuir a dimensão dos dados desnecessários num sistema e definir um modelo de gestão mais apropriado para os dados mantidos nos referidos sistemas.

Keywords: Bases de dados; Governação de Dados; Gestão de dados; Mineração de Dados; Aprendizagem Máquina.

Abstract

It is very likely that most information systems have at short-term information management problems. This requires the creation of new types of data management techniques more efficient and specific to each case, with the capacity to govern and ensure compliance with the management measures defined for operational systems, and ensure the desired performance and quality. In this work, we address the problem of data management, and, using a solution based on machine learning techniques, we tried to perceive, learn and classify the data contained in any database, according to its relevance for the users. Being able to identify what is really important to the users and separate this information from the rest, it is a great way for reducing the size of unnecessary data in a system and to define a more appropriate management model for the data that must be maintained in the system.

Keywords: *Databases; Data Governance; Data Management; Data Mining; Machine Learning.*

1. INTRODUÇÃO

As bases de dados são arquitetadas para guardar, organizar e fornecer de forma rápida e eficiente a informação necessária às atividades de cada organização, de tal forma que a maior parte destas atividades não seriam possíveis de se realizar sem o apoio de tais sistemas (Connolly e Begg, 2005). Dada a situação, é fácil de prever os efeitos desta dependência. O excesso de informação e

ISSN 2183-489X
DOI http://dx.doi.org/10.18803/capsi.v17.345-353

as quebras de desempenho dos sistemas de informação são apenas dois aspetos que poderão ocorrer como resultado de uma ineficiente gestão de dados. Ao contrário do que muitos possam imaginar, se analisarmos a corrente atual do desenvolvimento tecnológico, o problema de gestão de dados na maioria das situações não está desvanecido, bem pelo contrário, o problema aumentará dado o atual crescimento exponencial do volume de dados que se prevê que continue (Zhu et al., 2009). Dado o impressionante aumento do volume de dados que se tem vindo a registar, sem uma solução adequada para a sua gestão será muito difícil acomodar essa informação continuando com o estado de estabilidade atual na forma como se utilizam os sistemas de informação (Gantz e Reinsel, 2012). Contudo, tem-se verificado que todas as atividades geram uma quantidade enorme de dados, que ultrapassam em muito os limites convencionais para uma gestão eficiente, devido à falta de adaptação ao paradigma atual dos sistemas de informação e à sua má gestão. Como tal, muitas das soluções nesse domínio revelam-se desajustadas no contexto atual (Grolinger et al., 2013) (Marr, 2016b).

Analisando o problema, bem como as suas causas, é possível pensar em algumas soluções, nas quais sobressai a hipótese de se conseguir identificar os dados que são realmente importantes para um dado sistema, distinguindo-os dos demais. Posto isto, a solução que aqui expomos tem como objetivo principal identificar e distinguir a informação contida num sistema de bases de dados de acordo com a sua relevância para os utilizadores do sistema. Isto porque se acredita que ao classificar a informação com base num critério de importância para o utilizador permitirá definir aquilo que é de interesse e com qualidade para a organização e para os seus utilizadores. Tendo a informação classificada, cada administrador poderá selecionar as melhores medidas para gerir os dados nos sistemas que administra e reforçar as políticas de gestão de dados definidas *a priori*.

2. TRABALHO RELACIONADO

Os investigadores no domínio dos sistemas de informação têm vindo a desenvolver novos esforços para conseguir otimizar os novos sistemas, contribuindo para o surgimento de novos processos e técnicas de otimização, tanto a nível operacional como de gestão da informação, o que, obviamente, deve ser incentivado (Connolly e Begg, 2005). Quanto à gestão de informação, esta considera todos os procedimentos, tarefas e deveres que devem ser definidos e implementados num dado sistema para garantir a sua qualidade. Acredita-se que, se estas tarefas forem implementadas e bem orquestradas num sistema, então seremos capazes de garantir a qualidade, escalabilidade, performance, disponibilidade, segurança e consistência dos dados (Sakr et al., 2011), sendo que, algumas das tarefas que estão usualmente incluídas no domínio da gestão de dados são: a redução da dimensão dos dados, a detecção de anomalias e inconsistências, a definição de estratégias de dados, a monitorização de dados, a compreensão do negócio, entre muitos outros.

A nível operacional, as técnicas de otimizações de bases de dados devem também ser exploradas para mitigar os problemas dos sistemas. Alguns exemplos deste tipo de técnicas, são a inserção de índices para aumentar a velocidade das chaves primarias, ou a criação de grupos e ordenação dos dados por critérios de preferência ou de utilização. Outras técnicas mais avançadas foram também postas em prática, tais como novas formas de especificar o método de como e onde a informação é guardada, por exemplo, manter a informação nos discos rígidos ou noutras estruturas de memória para aumentar a disponibilidade ou velocidade de resposta do sistema. Outros métodos, mais refinados, também podem ser encontrados na literatura, como é o caso da estimação de custos de processamento de determinados pedidos, da avaliação de planos de execução de *queries* e até da implementação de planeadores de inquirições, que são responsáveis por orquestrar a execução das operações num sistema da forma mais eficiente possível. Mais recentemente, outras ideias de otimização emergiram de outras áreas relacionadas, as quais podem também ser utilizadas em sistemas de informação ditos convencionais. Referimo-nos, por exemplo, a métodos e técnicas relacionadas com a desnormalização de tabelas (Kimball, 1996), a materialização de vistas (Ioannidis, 1996) ou a agregação de dados e a seleção de vistas (LaBrie & Ye, 2002).

A solução que propomos apoia-se num processo de análise e de classificação realizado através de técnicas de captura de comportamentos numa base de dados para alimentar um motor de *machine learning*. Os resultados gerados permitem reforçar as políticas de gestão de dados definidas com algum impacto, uma vez que funcionará como apoio à gestão de informação como complemento a todos os outros mecanismos já instalados num sistema. Este tipo de solução poderá, contudo, não ser a mais indicada para todos os casos, uma vez que apresentará sempre uma percentagem de erro associada aos seus resultados. Porém, é uma solução diferente, de aplicação genérica e adaptável a qualquer sistema de bases de dados com os problemas apresentados.

3. PROCESSAMENTO E CLASSIFICAÇÃO

A solução desenvolvida permite analisar qualquer base de dados. Através da aplicação desta solução, defende-se que se poderá aumentar a qualidade de um sistema em termos de desempenho e gestão de dados, quando se identifica aquilo que é ou não relevante para o sistema. Adquirindo esse conhecimento, tem-se a possibilidade de reforçar as estratégias de gestão de dados definidas. A forma como a solução é aplicada e a maneira como os seus processos são realizados estão apresentadas na Figura 1. Analisando um pouco melhor o esquema de processamento, é possível identificar as 3 fases principais da solução e todos os seus subprocessos, que são, nomeadamente: a seleção e extração dos dados para análise, o processamento da informação pelo módulo de *machine learning* e por fim, a aplicação das medidas de gestão de dados.

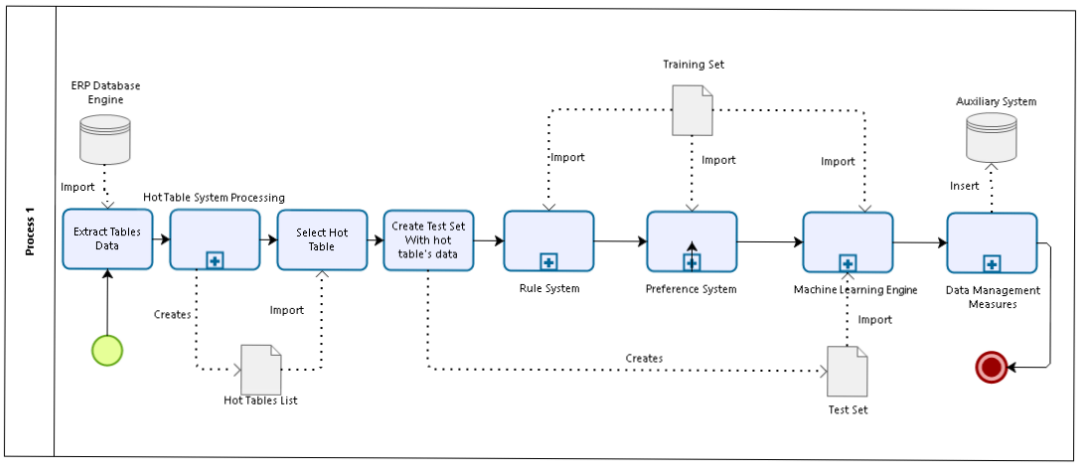

Figura 1 – Esquema geral do processamento da solução definida.

Para implementarmos a solução encontrada optámos por utilizar um método de classificação supervisionada. Este tipo de método requer que se forneça ao algoritmo de classificação algum conhecimento prévio, de forma a que este possa criar um modelo capaz de aferir a relevância dos diversos elementos de dados contidos nos sistemas. Portanto, o processo de extração de conhecimento implementado utiliza os registos operacionais do sistema, pois fornecem informações cruciais sobre as operações do sistema e, com isso, saber quais as tabelas, atributos e valores que mais relevo têm no sistema. A atribuição do grau de relevância é feita através do tipo de operação e é representado por um novo atributo em cada conjunto de treino – a classe, que é multinominal, poderá ter tantos níveis quantos aqueles que o utilizador desejar. Por exemplo, se tratasse de uma operação de inserção, então seria bastante provável que essa operação interessasse e que fosse anotada como tal. No caso contrário, teríamos, pelo motivo contrário, uma operação de remoção. Contudo, existem outros fatores que devem ser equacionados, como, por exemplo, o peso de cada atributo e a operação realizada. Como este é um método que pode não refletir uma configuração adequada da utilização dos dados, é necessário refinar os conjuntos de treino ao capturar mais conhecimento. Quanto aos dados de teste, estes são obtidos da extração das instâncias de uma tabela.

Sabendo-se que cada utilizador tem sempre preferências sobre certos dados, projetou-se e implementou-se um sistema de refinamento com capacidade para atuar sobre os conjuntos de dados de treino, de acordo com as preferências de utilização dos dados conhecidas, tendo como referência alguns trabalhos realizados no domínio do processamento analítico de dados (Golfarelli e Rizzi, 2009). Assim, é possível ajustar o nível de relevância definido inicialmente para os conjuntos de treino. Com a incorporação destes sistemas foi possível aumentar o conhecimento obtido dos utilizadores e, por sua vez, melhorar a eficiência dos métodos de classificação. Relativamente à escolha dos algoritmos para realizar esta operação, esta incidiu sobre um dos algoritmos clássicos, o *Naive Bayes*, devido às vantagens que este nos apresenta para classificação

textual. Por fim, testámos uma implementação de um algoritmo de *Deep Learning*, bastante inovador, tendo em consideração o seu desempenho em situações nas quais existe pouco conhecimento para treino dos modelos de classificação. No primeiro, optou-se pela implementação da plataforma *Weka* em *Java* e, no segundo, pela implementação disponibilizada pela plataforma *H2O* em *R* (Ahmed & Jesmin, 2014) (Candel et al., 2017).

Para que a eficiência dos métodos de *machine learning* fosse garantida foi necessário fazer uma preparação dos dados adequada e adaptá-los ao algoritmo a utilizar. Assim, é possível prever, à partida, o problema de que diferentes tabelas terão tipos de dados diferentes e, como tal, preparações também distintas (Zhang et al., 2003). Para que fosse possível produzir uma abordagem genérica e adequada à maior parte dos casos, implementámos um conjunto de procedimentos fixos para aplicável a qualquer conjunto de dados. Entre as tarefas principais desses procedimentos podem-se destacar a limpeza e a conformação de dados, a discretização de atributos numéricos, a eliminação de *outliers*, a normalização de inteiros, a eliminação de atributos com variações pequenas nos seus valores, entre outros processos básicos de preparação.

Para que fosse possível reduzir o domínio de análise e escolher as melhores tabelas de um sistema a serem submetidas ao procedimento, projetou-se e implementou-se um sistema de seleção de tabelas inspirado numa solução de escolha de vistas OLAP (Rocha e Belo, 2015). Este sistema de seleção analisa os registos de uma base de dados e as operações que cada utilizador realiza numa sessão do sistema através de uma cadeia de *Markov*. A cadeia é um grafo de tabelas e suas ligações refletindo o uso do sistema por cada sessão, com o objetivo de se encontrarem as mais influentes. O sistema distingue a importância das tabelas através de cores segundo três critérios. O primeiro é a taxa de ocupação mínima para definir qual o espaço mínimo que cada tabela deverá ter (o que elimina as tabelas mais "pequenas"), a taxa de uso mínima para triar as tabelas mais utilizadas das restantes e por fim, o critério de máximo espaço que uma tabela poderá ocupar. A cor atribuída a cada nodo é definida com base no peso desses critérios. Cores mais fortes significam um maior rigor nos critérios aplicados, enquanto que cores mais claras indicam elementos menos adequados. Por fim, procede-se à eliminação das ligações mais fracas – aquelas que têm probabilidades mais reduzidas –, e dos nodos isolados, ou seja, elementos que não tenham qualquer ligação entre o início e o fim de uma dada sessão. Os restantes nodos representam as tabelas do sistema que são mais indicadas para submeter ao processamento da solução.

4. ANÁLISE DO CASO DE ESTUDO

Para fazer a validação da solução que propomos, projetámos e implementámos um ambiente específico. Para isso, utilizámos como base de trabalho uma base de dados com alguma dimensão de um *Enterprise Resource Planning* (ERP), integrando dados relativos à gestão de um polo universitário, que estão armazenados em mais de *1000* tabelas, contendo alguns milhões de

registos – aqui colocou-se em causa a escalabilidade do sistema, além dos típicos problemas de qualidade de dados que podem ocorrer.

Inicialmente, definiu-se a estratégia para a realização da avaliação e decidiu-se que seria feita uma execução geral ao sistema de informação completo com a finalidade principal de se simular e avaliar o tempo de execução total e um teste individual a uma tabela, com a finalidade de se analisar a possível redução de espaço resultante das classificações inferidas. O sistema capturou mais de *1,700,000* registos de operações, podendo-se identificar no seu conjunto *133* tabelas diferentes que serão processadas. Este número de registos revela um volume de dados de utilização do sistema de pequena dimensão. Apenas existem registos para algumas tabelas do sistema. Portanto, através destes registos não será possível ter informação sobre a maior parte das estruturas. Porém, isso não é um problema, uma vez que significa que as restantes tabelas não são relevantes à partida ou que são apenas de leitura para apoio à aplicação, não tendo sido registadas as operações que sobre elas foram realizadas. No teste individual, foram especificados os resultados para a tabela de movimentos, relativa a transações monetárias do polo universitário, pois se reconheceu ser uma das tabelas mais críticas e interessantes do sistema, existindo mais de *1,000,000* de registos operacionais apenas desta tabela, o que confere um grau de fiabilidade elevado aos resultados gerados para esta tabela. Em suma, avaliou-se o tempo de execução do procedimento, a percentagem de espaço que potencialmente se poderia ganhar ao eliminar (ou descentralizar) a informação elegível dos sistemas principais para estruturas auxiliares (históricas ou *cache*). Estas foram as medidas que foi possível aplicar.

5. ANÁLISE DOS RESULTADOS OBTIDOS

Na execução geral, no processo de seleção das tabelas na cadeia de *Markov* foram definidos os critérios de mínimo uso, com um valor de *15%* para a taxa de utilização, de *70%* e *5%* para o espaço máximo e mínimo, respetivamente, que a tabela teria de ocupar para ser selecionada. O peso de cada um destes critério foi, respetivamente, de *30%*, *65%* e *5%*. Da seleção resultante, foram escolhidas cerca de *15%* (*20*) das tabelas com registos, o que reduziu bastante a dimensão do processo de análise. Estas configurações foram escolhidas uma vez que achámos serem as mais adequadas para a dimensão da base de dados, bem como para a quantidade de registos operacionais disponível.

Uma vez terminado o procedimento, teve-se um tempo de processamento total de cerca de *2* horas e *3* minutos, o que perfez uma média de *6.15* minutos para cada tabela escolhida. A percentagem de espaço médio que poderia ter sido poupada com a aplicação de medidas de gestão (descentralizar ou remover) é de cerca de *74%* do espaço total da base de dados, antes do processamento da solução, que engloba as entradas com o máximo e o mínimo de relevância, ou seja, as diferenciadas pelo uso do sistema. *9.43*% foi a percentagem média obtida relativa às

entradas classificadas como não relevantes, ou seja, aquelas que podem ser eliminadas com maior segurança, que representam uma fração mais realista do que se poderia ganhar em termos de espaço. A precisão média de previsão dos algoritmos, resultante da *cross-validation* dos dados de treino, foi bastante satisfatória bem como o tempo de execução (Figura 2), sabendo-se que a base de dados tem uma dimensão de cerca de *4 GB*. Uma das razões para o tempo não ter sido muito elevado, deve-se ao facto de o algoritmo de *deep learning* ter apenas sido requisitado em 10% (2) das vezes, o que evitou o processamento pesado na maior parte dos casos. Porém, aumentou a qualidade das previsões.

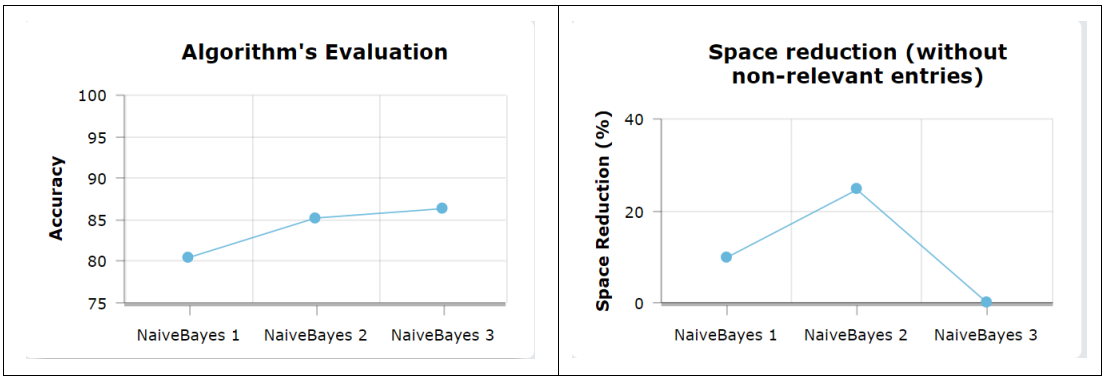

Figura 2 – Avaliação do algoritmo e representação do espaço poupado pela eliminação de dados irrelevantes.

Analisando os resultados relativos ao processamento da tabela de movimentos, foi apenas necessário processar os dados através do algoritmo de *Naive Bayes*, dada a sua elevada precisão, perfazendo um total de *34.36* minutos em média por cada processamento dos *3* testes realizados. Quanto à precisão do algoritmo, obteve-se um valor médio de *85.64%* de precisão analisando os resultados de um processo de *cross-validation* com o conjunto de treino. Nestes testes, as preferências e as regras foram diferentes para as três execuções, tendo sido definidas manualmente (ao contrário do teste anterior) através da observação dos valores e suas variações, com a ajuda de uma ferramenta analítica. Analisando os resultados e a percentagem de espaço que poderia ser poupado, conclui-se que este está dentro da média global, podendo-se poupar cerca de *11.28%* com a descentralização dos dados irrelevantes (os de máxima relevância não incluídos) do sistema principal. Na Figura 2 pode-se observar a avaliação dos algoritmos para a tabela de movimentos, bem como um gráfico com a percentagem de espaço poupado nas execuções realizadas.

O espaço libertado nos diferentes casos é diferente, isto deve-se às diferentes preferências usadas no processo que influenciam os modelos preditivos gerados distintamente. A percentagem total de entradas distinguidas pelo processamento, foi de cerca de 85%. Este é um número bastante interessante, pois significa que é possível dar um novo significado à maior parte dos dados contidos na tabela em questão. O nível verificado para a qualidade dos resultados deve-se ao facto

de existirem bastantes registos de operações para esta tabela e à escolha das preferências definidas inicialmente.

6. CONCLUSÕES E TRABALHO FUTURO

Com base na interpretação dos resultados obtidos é possível concluir que estes são satisfatórios e esclarecedores. Apesar da dimensão da base de dados não ser a desejada, foi possível realizar os testes individuais sobre uma tabela com grande dimensão, que continha uma quantidade de registos de utilização bastante interessantes, conferindo um grau de fiabilidade bastante aceitável aos resultados das classificações. Para o teste da execução geral, achamos que foi possível ter uma ideia bastante concreta de quanto tempo um procedimento deste género poderá demorar num caso real com esta magnitude. No caso de estudo, verificou-se que havia um uso muito acentuado de uma tabela relativamente a todas as outras, o que reduziu o impacto prático que os testes realizados poderiam ter em termos de optimização de espaço. Apesar disso, o exemplo é bastante ilustrativo das capacidades da solução definida. Da análise aos testes da tabela singular é possível prever como seriam os resultados noutros casos de aplicação, se a quantidade de registos (operacionais) e entradas na tabela for proporcionalmente semelhante ao utilizado nos teses realizados. Um fator determinante que foi identificado para o sucesso da solução foi a captura de conhecimento de utilização do sistema por parte dos seus utilizadores. Assim, uma possível forma para aumentar o desempenho do procedimento seria criar novas maneiras para capturar tais dados, para além dos registos das operações na base de dados, por exemplo, um mecanismo de captura de ações dos utilizadores e uma forma de fazer a sua tradução.

Quando à aplicação dos resultados, esta é provavelmente a parte mais entusiasmante de toda a solução, dado o seu impacto prático no sistema. Com as medidas de descentralização enunciadas, é possível aumentar o desempenho no processamento de algumas das tabelas do sistema. Contudo, outras medidas podem ser descobertas através de análises específicas sobre os dados classificados. Por exemplo, se a partir da análise dos conjuntos de dados surgirem elementos pertinentes acerca dos dados de uma tabela, então poder-se-á preparar medidas mais específicas para o tratamento dessa mesma informação. De facto, podemos defender que ao se ter conhecimento daquilo que é importante para o utilizador de um sistema, implicitamente está-se a definir a qualidade da informação para esse sistema e, assim, melhorar consideravelmente a gestão dos seus dados.

REFERÊNCIAS

Ahmed, K., & Jesmin, T. (2014). Comparative Analysis of Data Mining Classification Algorithms in Type-2 Diabetes Prediction Data Using WEKA Approach. *International Journal of Science and Engineering*, *7*(2). https://doi.org/10.12777/ijse.7.2.155-160

Candel, A., LeDell, E., Parmar, V., & Arora, A. (2017). *Deep Learning with H2O - Booklet*. H2O.ai, Inc.

Connolly, T. M., & Begg, C. E. (2005). *Database Systems: A Practical Approach to Design, Implementation, and Management*. Pearson Education.

Gantz, J., & Reinsel, D. (2012). *THE DIGITAL UNIVERSE IN 2020: Big Data, Bigger Digital Shadows, and Biggest Growth in the Far East*. IDC.

Golfarelli, M., & Rizzi, S. (2009). Expressing OLAP Preferences. In M. Winslett (Ed.), *Scientific and Statistical Database Management* (Vol. 5566, pp. 83–91). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://link.springer.com/10.1007/978-3-642-02279-1_7

Grolinger, K., Higashino, W. A., Tiwari, A., & Capretz, M. A. (2013). Data Management in Cloud Environments: NoSQL and NewSQL Data Stores. *J. Cloud Comput.*, *2*(1), 49:1–49:24. https://doi.org/10.1186/2192-113X-2-22

Ioannidis, Y. E. (1996). Query optimization. *ACM Computing Surveys*, *28*(1), 121–123. https://doi.org/10.1145/234313.234367

Kimball, R., & Ross, M. (2013). *The data warehouse toolkit: the definitive guide to dimensional modeling* (3rd edition). Indianapolis, Ind: Wiley.

LaBrie, R., & Ye, L. (2002). A Paradigm Shift In Database Optimization: From Indices To Aggregates. 5.

Marr, B. (2016). *Big Data Overload: Why Most Companies Can't Deal With The Data Explosion*. Forbes. Retrieved from http://www.forbes.com/sites/bernardmarr/2016/04/28/big-data-overload-most-companies-cant-deal-with-the-data-explosion/#62164c203920

Rocha, D., & Belo, O. (2015). Integrating usage analysis on cube view selection - an alternative method. *International Journal of Decision Support Systems*, *1*(2), 228. https://doi.org/10.1504/IJDSS.2015.067559

Sakr, S., Liu, A., Batista, D. M., & Alomari, M. (2011). A Survey of Large Scale Data Management Approaches in Cloud Environments. *IEEE Communications Surveys Tutorials*, *13*(3), 311–336. https://doi.org/10.1109/SURV.2011.032211.00087

Zhang, S., Zhang, C., & Yang, Q. (2003). Data preparation for data mining. *Applied Artificial Intelligence*, *17*(5–6), 375–381. https://doi.org/10.1080/713827180

Zhu, Y., Zhong, N., & Xiong, Y. (2009). Data Explosion, Data Nature and Dataology. In N. Zhong, K. Li, S. Lu, & L. Chen (Eds.), *Brain Informatics: International Conference*, BI 2009 Beijing, China, October 22-24, 2009 Proceedings (pp. 147–158). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://dx.doi.org/10.1007/978-3-642-04954-5_25